OOP Forum Week 9: Inheritance

Number 1

1. Inheritance
2. Association
3. UML Diagram :

|  |
| --- |
| Species |
| - String: speciesName + setSpeciesName(String s) + String getSpeciesName() + String toString() + boolean equals(Species s) |

1. Avoids code duplication and simplifies testing.
2. Because the toString() method is related to the class it is declared in. the compiler will select the correct method / function.
3. Polymorphism

Number 2

1. The practice of hiding data / making the fields in a class private and providing access to the fields via public methods ( accessor / mutators )
2. Ease of testing and gives maintainability, flexibility, and extensibility to code.
3. private String getName()
4. private String name
5. \*in code\*
6. Advantage: inherits all of the attributes of the species class.  allow code in specimen object to access species-related methods directly. Disadvantage: Inconsistent data.  data in the species class may not be consistent across the associated Specimens.

Number 3

* 1. Description of each markings can be inputted into an instance variable within the specimen class. toString() would be updated to include the description. They should also include getter and setters to update the markings as they age.
  2. \*in code\*
  3. //ANIMALS is a unique array containing Specimen objects

//UNIQUE is a collection, initially empty

//SPECIES is a Species object

loop BEAST from 0 to the number of elements in ANIMALS

NEW\_SPECIES = true

UNIQUE.resetNext()

loop while UNIQUE.hasNext()

if UNIQUE.getNext() = species of

ANIMALS[BEAST] then

NEW\_SPECIES = false

end if

end loop

UNIQUE.resetNext()

loop while UNIQUE.getNext()

SPECIES = UNIQUE.getNext()

output SPECIES.toString()

end loop

Number 4

1. ADT is a type for objects whose behaviour is defined by a set of value and a set of operation. ADT exports a type. It also exports a set of operations. This set is called interface. Operations of the interface are the one and only access mechanism to the type's data structure. It is called “abstract” because it gives an implementation-independent view. The process of providing only the essentials and hiding the details is known as abstraction.
2. \*in code\*
3. \*in code\*
4. \*in code\*